

UNIVERSIDAD
ICESI

Computación y estructuras computacionales I

Tarea Integradora III

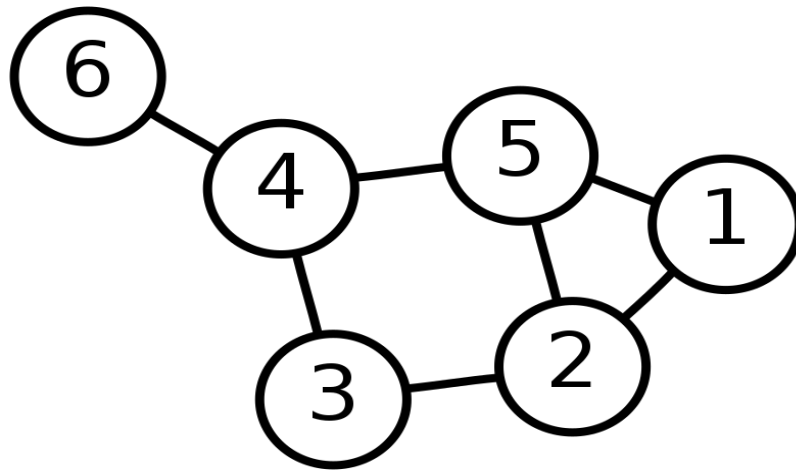
TAD - Estructura : Grafo

TAD...¿Qué es?

Debemos recordar que el TAD es una herramienta eficaz que permite la modelación de las operaciones que deben realizar un conjunto de datos. Gracias a el, se puede llegar a una definición formal de lo que debemos modelar (hablando desde el punto de los algoritmos), por está razón, hemos decido construir el TAD que le corresponde al grafo que hemos implementado para nuestro proyecto.

Nombre: Grafo / Graph

Objeto abstracto:



$G = (V, E, directed, weight), V = \{v_1, v_2, v_3, \dots, v_n\} \wedge E = \{e_1(v_{i1}, v_{j1}, w_1), e_2(v_{i2}, v_{j2}, w_3), e_3(v_{i3}, v_{j3}, w_3), \dots\}$

Invariante:

$\forall e_i \in E \wedge \forall e_j \in E, e_i(v_{i1}, v_{j1}, w_i) \neq e_j(v_{j1}, v_{i1}, w_j)$

Operaciones: Las operaciones primitivas que comprende la estructura de grafo son:

- create: $\langle \text{Vertex}, \text{Graph} \rangle \rightarrow \text{Graph}$ < Constructor >
- addVertex: $\langle \text{Vertex}, \text{Graph} \rangle \rightarrow \langle \text{boolean} \rangle$
- addEdge: $\langle \text{Vertex1}, \text{Vertex2}, \text{weight}, \text{Graph} \rangle \rightarrow \langle \text{boolean} \rangle$
- deleteVertex: $\langle \text{Vertex}, \text{Graph} \rangle \rightarrow \langle \text{boolean} \rangle$
- searchVertex: $\langle \text{Vertex}, \text{Graph} \rangle \rightarrow \langle \text{boolean} \rangle$

Operaciones

Teniendo clara la definición de la estructura y que operaciones debe realizar, pasaremos a describir cada una de estas.

Nombre: Create
Descripción: Operación de creación. Se hace una instancia de un grafo.
Precondición: N/A
Postcondición: Grafo creado, su estado es vacío. $G = \{V\{\emptyset\}, E\{\emptyset\}\}$

Nombre: addVertex <Vertex, Graph>
Descripción: Operación de agregar vértice. Se agrega un vértice al grafo creado
Precondición: $v \notin V$.
Postcondición: Se agrega el vértice al conjunto de vértices del grafo. $v \in V$.

Nombre: addEdge: <Vertex1, Vertex2, weight, Graph>
Descripción: Se agrega una arista que conecta dos vértices al grafo.
Precondición: $v_1 \in V \wedge v_2 \in V$.
Postcondición: Se agrega una arista al grafo. $G = \{V = \{v_1, v_2\}, E = \{e_1(v_1, v_2, w_1)\}, directed, weight\}$.

Nombre: deleteVertex <Vertex, Graph>
Descripción: Elimina un vértice del conjunto de vértices del Grafo. Además de eliminar toda asociación que se tenga.
Precondición: $v \in V$.
Postcondición: $v \notin V \wedge v \notin E$.

Nombre: searchVertex: <Vertex, Graph>
Descripción: Se busca un vértice en el conjunto de vértices del grafo.
Precondición: N/A
Postcondición: $(v \in V) \rightarrow true \vee (v \notin V) \rightarrow false$.